Synopsis

* 1. **Title of the project**

AWAAZ

**1.2 Objective of the project**

• To support physically liable people.

• To convert speech to text and text-to-Speech.

• To convert Text-to-Speech using Image, Pdf and Doc.

* 1. **Project Category**

Desktop Application

**1.4 Language(s) to be used**

Frontend: Kivy

Backend: Phyton

**1.5 Structure of the program**

**1.5.1 Analysis**

Having difficulties like being visually impaired, dumb are greater amount of concern. Science and innovation have influenced people to turn out to be dependent on solace yet there exists an underprivileged gathering of individuals who are battling for finding a creative way that can make the procedure of communication simpler for them. Conversations between people who lack the ability to talk and hear with a normal person has always been a challenging task. We must be treated as equal and conversation is the way this can be brought about. There are about 12.3 million people in India with moderate to complete hearing loss and there are 478 schools receiving government funding and approximately 372 private schools for the physically liable scattered throughout India. This is an application which aims to solve the problem of people with vision, hearing and speech impairment called a deaf-mute-blind communication system using Machine Learning. The app has features like voice assistant and converting it into text, OCR text detection using Image, Pdf, Document and conversion of text to audio to texts.

**1.5.2 Module Descriptions**

**1.5.2.1 Register**

In this new user will register to the system using user credentials username, password and User will get a Password Recovery Key.

**1.5.2.2 Login**

In this already registered user can login to the system using username and password.

**1.5.2.2.1 Logout**

By this module user can logout from the Application.

**1.5.2.3 Forgot Password**

In this user can change their password if they have forgot their login password.

**1.5.2.4 Image To Text**

In this user can select a image having text using the file manager or the file explorer that will automatically converted into Text and after user can convert it to voice and also user can copy this text.

**1.5.2.5 Pdf To Text**

In this user can select a pdf having text using the file manager or the file explorer that will automatically converted into Text and after user can convert it to voice and also user can copy this text.

**1.5.2.6 Doc To Text**

In this user can select a Document having text using the file manager or the file explorer that will automatically converted into Text and after user can convert it to voice and also user can copy this text.

**1.5.2.7 Voice To Text**

This module listen to the users voice and converts it to the text, and user can copy this text.

**1.5.2.8 Text To Voice**

In this user can input text using the keyboard and it will be converted into voice

**1.5.2.9 Theme**

**1.5.2.9.1 Style**

This module will be used to change the theme style

i.e. Light or Dark.

**1.5.2.9.2 Color**

This module will be used to change the primary color of the application

i.e. The color of Buttons, menus etc.

**1.5.2.10 Voice Assistant**

**1.5.2.10.1 Record**

This module is used to listen to the user and to record the user voice.

**1.5.2.10.2 Check**

This module is used to check the semantics in the converted voice.

**1.5.2.10.3 Response**

This module is used to find the proper response for the user query.

**1.5.2.10.4 Speak**

This module is used to deliver the response to the user by voice.

**1.5.2.11 Help**

This module describes about the application and there will be instruction about each of the module.

**1.6 Data Structure**

Not Applicable.

**1.7 Any Other Information**

Not Applicable.

**1.8 Future Scope**

This project can be improvised by adding different languages and upgrading the modules by more sophisticated ml modules.

**SOFTWARE REQUIREMENT SPECIFICATION**

**2.1 Introduction**

A Software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. It also describes the functionality the product needs to fulfill all user needs.

A software requirement specification is a description of software system to be developed. It has to bridge communication gap between developer and user. It describes functional and non-functional requirement and may include a set of cases that describes user interaction that the software must provide.

A typical SRS includes:

• A Purpose

• An overall description

• Specific requirement

**2.1.1 Purpose**

The purpose of this document is to provide overall description of the event with functionalities and their dependencies with each other and their requirement.

An SRS forms the basis of an organization’s entire project. sets out the framework that all the development teams will follow. It provides critical information to all the teams, including development, operations, quality assurance (QA) and maintenance, ensuring the teams are in agreement.

Using the SRS helps an enterprise confirm that the requirements are fulfilled and helps business leaders make decisions about the lifecycle of their product, such as when to retire a feature.

In addition, writing an SRS can help developers reduce the time and effort necessary to meet their goals as well as save money on the cost of development.

**2.1.2 Scope**

Boundaries of software products are defined by set of requirements. The software development team designs, implements, tests and 8 deliver this requirements to user. A requirement is automatic unit of software product.

SRS document provides a reference for validation of the final product. A high quality SRS is a prerequisite to high quality software. In future one can refer the following SRS to extend the same software by referring this SRS document other developers can develop and extend this software to network level. Is specifies basic requirement constraints and interface for the proposed system.

**2.1.3 Definitions , Acronyms , Abbreviations**

GUI – Graphical User Interface

SRS-Software of Requirement Specification.

**2.1.4 Reference**

• An Integrated Approach to Software Engineering – Pankaj Jalote.

• Synopsis.

• URL-https://www.perforce.com/blog/alm/how-write-software-requirementsspecification-srsdocument#:~:text=Why%20Use%20an%20SRS%20Document,assurance% 2C%20operations%2C%20and%20maintenance

**2.1.5 Overview**

Software requirement specification gives overview of the functionality of the software. SRS establishes the basic or agreement between user and developer can what the software product will do. SRS helps to understand their own needs. The SRS document describes the various system requirements, interfaces features and functionality in details.

**2.2 Overall Description**

This section describes the function of the project and their aim. It Also includes the constraints and the requirements of the project.

**2.2.1 Product Perspective**

Product Perspective is an essentially the relationship of the product to other products defining whether the product is independent or is a part of longer product function.

This project is mainly designed to support physically liable people. And it acts as an virtual assistant to the liable people.

**2.2.2 Product Function**

The main function of the project is to cover the communication gap between the physically liable people and the normal people. This project will create a different way of communication.

**2.2.3 User Characteristics**

User should have basic computer knowledge and know about functionalities of the application and user should be able to read or understand English.

**2.2.4 General Constraints**

This product developed using python language. It operates on Windows 7 and higher version and it needs to have 4GB RAM and 500GB Hard Disc.

**2.2.5. Assumption and Dependencies**

The basic assumption is made about end users that they should be computer literate and should know the usage of computer system and should know English.

**2.3 Specific Requirement**

The specific requirement section describes all the details that the software developer needs to know for designing and developing the software.

**2.3.1 External Interface Requirement**

The external requirements are types of functional requirements. And they outline how our product will interface with other components, This section describes all the interface of the software to people, other software, hardware and other system.

**2.3.1.1 User Interface**

The graphical user interface makes application more interactive which includes buttons and labels for selecting options and perform operations.

**2.3.1.2 Hardware Interface**

To run this app the basic system requirements are as follows: Hard Disk -500 GB, 4GB RAM and Input Device such as Keyboard, Mouse etc.

**2.3.1.3 Software Interface**

This product uses python 3.9.2 and kivy 2.0 and it operates an window 7 and other higher versions.

**2.3.2 Functional Requirement**

Functional requirements are in an SRS document indicate what a software system must do and how it must function; they are product features that focus on user needs.

**2.3.2.1 Register**

In this new user will register to the system using user credentials username and password.

**Input:** username, password.

**Process:** Check username, password against different validation if both username and password are valid user will be registered and a security key will be generated.

**Output:** if both username and password are valid user will be registered and a security key will be generated and displayed to the user.

**2.3.2.2 Login**

**I**n this user can login to the system using username and password.

**Input:** username, password.

**Process:** Check for username, password

**Output:** If username, password are valid,then user will login to the system.

**2.3.2.2.1 Logout**

In this user can logout of the system using logout button.

**2.3.2.3 Forgot Password**

In this user can change their password if they have forgot their login password.

**Input :** Security Key

**Process:** When user enters security key, if the security key is valid, then user will be asked to enter the new password where user can change their password.

**Output:** Password will be changed

**2.3.2.4 Image To Text**

In this user can convert the image to text and then convert the text to voice.

**Input:** Upload Image with text from the system using the file manager or file explorer.

**Process:** Extract the text from the image using ocr and displays text from there user can also convert the text into voice

**Output:** Extracted text will be displayed, and voice output of the text also available.

**2.3.2.5 Pdf To Text**

In this user can convert the pdf to text and then convert the text to voice.

**Input:** Upload pdf with text from the system using the file manager or file explorer.

**Process:** Extract the text from the image using ocr and displays text from there user can also convert the text into voice

**Output:** Extracted text will be displayed, and voice output of the text also available.

**2.3.2.6 Doc To Text**

In this user can convert the Document to text and then convert the text to voice.

**Input:** Upload Document with text from the system using the file manager or file explorer.

**Process:** Extract the text from the image using ocr and displays text from there user can also convert the text into voice

**Output:** Extracted text will be displayed, and voice output of the text also available.

**2.3.2.7 Voice To Text**

In this user can convert the voice into the text format.

**Input:** User will give the voice input in the simple words user will just talk to the system.

**Process:** The inputted voice will be processed and converted into text format.

**Output:** The Converted text will be displayed

**2.3.2.8 Text To Voice**

In this user can convert the text into the voice format.

**Input:** User will give the input in the text format i.e. by using the keyboard.

**Process:** The inputted text will be processed and then it will be converted into voice.

**Output:** The converted voice will be played.

**2.3.2.9 Theme**

In this user can change the theme style and the theme color, there will be more than 500 colors are available and it can be set as primary color of the application.

**2.3.2.9.1 Style**

In this user can set the application theme i.e. light or dark.

**Input:** user will select the theme style i.e. light or dark.

**Process:** The selected color is checked that it is light or dark and the appropriate theme will be set.

**Output:** The theme will be changed to light or dark accordingly to the theme selected by the user.

**2.3.2.9.2 Color**

In this user can set the application default color.

**Input:** user will select the theme color from the theme selection window.

**Process:** The selected color is checked and that will be set as primary color of the application i.e. default color of the application.

**Output:** The default color of the application will be changed to the appropriate color selected by the user.

**2.3.2.10 Voice Assistant**

In this user can interact with the system using voice commands. i.e. by simply talking to the system user can do many things.

**2.3.2.10.1 Record**

In this module the user voice will be recorded for the further processing.

**Input:** Input will voice commands by the user.

**Process:** It will process the user voice and stored as text.

**Output:** The received voice is processedand returned as text for further processing.

**2.3.2.10.2 Check**

In this module the processed and converted text is used to search for the appropriate answer or to search for the appropriate command.

**Input:** The processed and converted user voice command is the input for this.

**Process:** The input will be further processed to find the appropriate command to execute or to find the answer for the user query.

**Output:** The found answer or command will be stored for further processing

**2.3.2.10.3 Response**

In this module the output of the check module is used as input and further processed to complete the task or execute the command and also it returns the success or failure message.

**Input:** The output of the check module is the input for this module.

**Process:** The system proceeds for the further execution it will execute the user command or resolves the user query and returns the failure or the success message to the user.

**Output:** The returned output or message will be stored for further processing.

**2.3.2.10.4 Speak**

In this module the output from the response module acts as the input for this module and the response will be converted to voice output.

**Input:** The output of the response module will acts as the input for this module.

**Process:** The input will be processed and it will be converted to voice and it will be delivered to the user.

**Output:** The converted voice output will be delivered to the user.

**2.3.2.11 Help**

This module describes about the application and there will be instruction about each of the module.

**2.3.3 Performance Requirements**

It uses Windows 7 and above version and 4GB RAM and 500GB Hard disk for its better performance.

**2.3.3.1 Static Requirement**

Static requirements are those that do not improve constraints on the execution characteristics of the system. These include requirements like number of terminals to be supported and the number of simultaneous users to be supported.

**2.3.3.2 Dynamic Requirement**

Dynamic requirement specifies constraints on the execution behavior of the system. These typically include response time and throughput constraints on the system.

**2.3.4 Design Constraints**

**2.3.4.1 Hardware Constraints**

The development of this application requires a computer system with 4GB RAM and 500GB Hard disk.

**2.3.4.2 Software Constraints**

This application development requires Python, Kivy and Windows 7 and higher version Operating System.

**2.3.4.3 Fault Tolerance**

Fault Tolerance is the property that enables a system to continue operating properly in the event of the failure of some of its components.

**2.3.4.4 Security**

If user gives invalid input the system displays the appropriate error message.

**2.3.4.5 Standard Compliance**

The software has Graphical User Interface (GUI) to ease of the user. Mouse and Keyboard interaction as per the standard software system.

**2.3.5 System Attribute**

**2.3.5.1 Maintainability**

This application will be developed in such a way that it can be modified.

**2.3.5.2 Availability**

This application is made available as Stand Alone Application.

**2.3.5.3. Reliability**

The application system must be highly reliable ands it should generate all the outputs in the correct order.

**2.3.6 Other Requirements**

Not applicable

**SYSTEM DESIGN**

**3.1 Introduction**

The design activity starts when SRS for software to be developed is available. It is the first step that we are moving from problem to solution domain. It is a blue print of plan for the system. It has two levels; First level is system or top level design which is the process of defining elements of system such as architecture modules and components. Different interface of those components and the data that goes through that system. It is a systematic approach to design or system.

**3.2 Applicable Documents**

* Synopsis
* SRS – Software Requirement Specification

**3.3 Functional Decomposition**

Functional decomposition is the process of taking a complex process and breaking it down into its simpler parts. Using functional decomposition large or complex functionalities decomposed into smaller parts that are more easily understood. It is mainly used during project analysis phase, so each phase can be viewed as software. So, this has modular with some sub module.